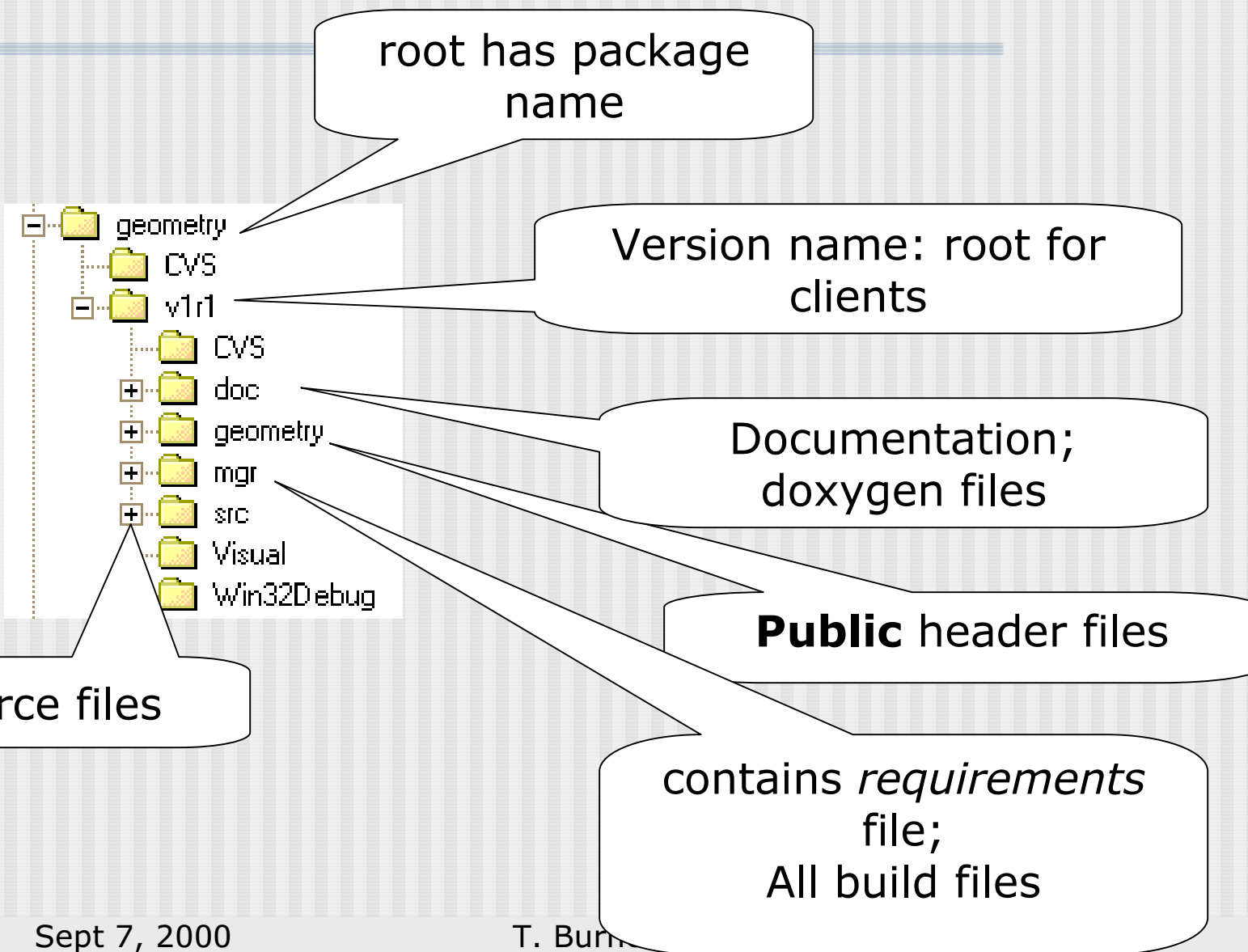


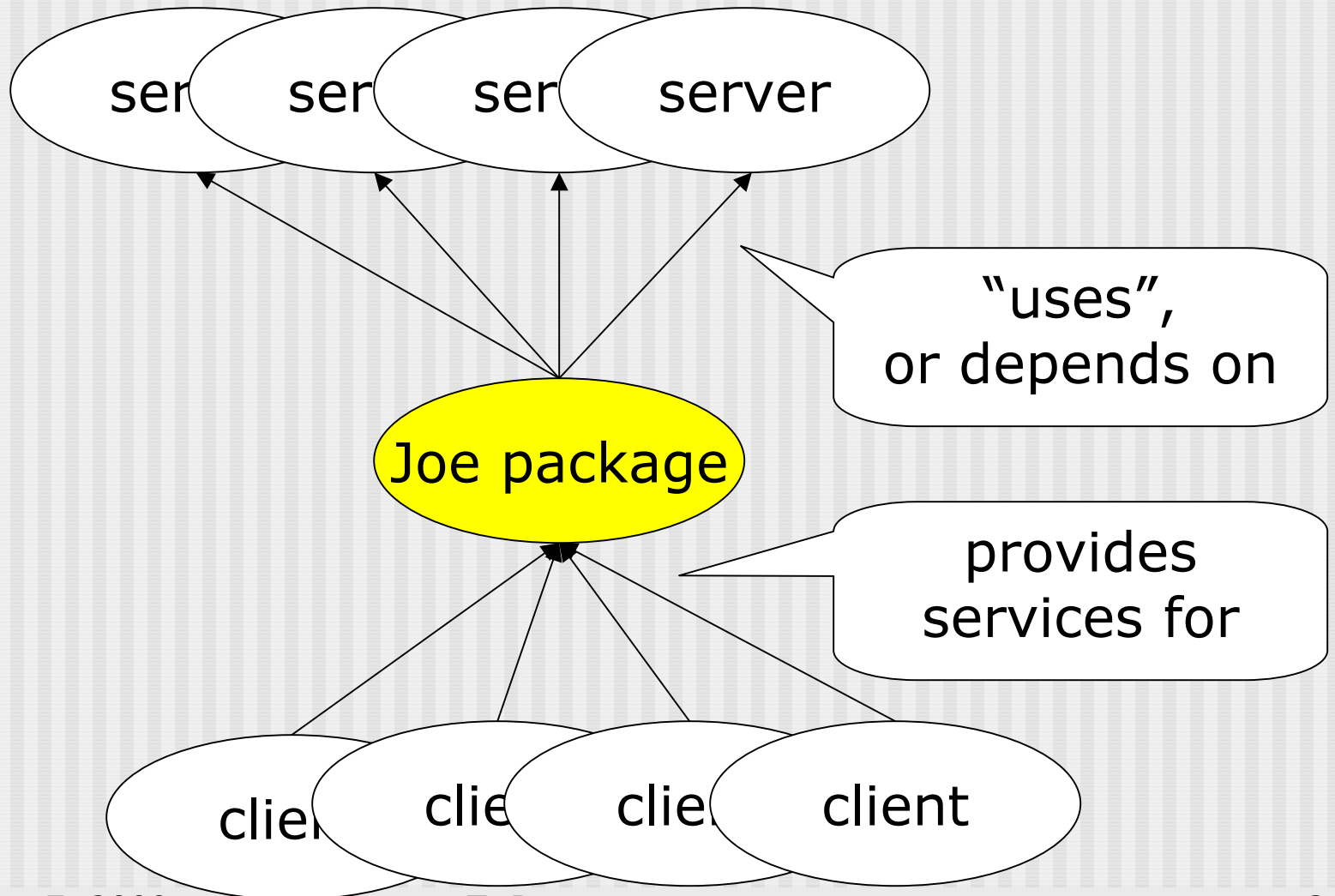
GLAST ground software package management: *CVS, CMT, doxygen, and vcmt*

Status, plans, issues;
vcmt demo.

A package: Example disk structure as defined by CMT



The package model: a tree



So what does "uses" mean?

- Version modifier
- Compile time dependencies
 - include paths
 - preprocessor defines
- Link time dependencies
 - Library paths, lib names
 - (so package must create its binaries, or *constituents*)
- Execution time dependencies
 - Shareable paths, names
 - Environment variable definitions, e.g., to data files
 - Alias names

About CMT (Configuration Management Tool)

- Special folder *mgr* (or *cmt*) contains two primary files:
 - *requirements*: complete description of interface
 - *Makefile*: standard + user-added targets
- Executable *cmt*: a parser of this file allowing:
 - query
 - building other files
 - make
 - DevStudio
 - any “document”
- References
 - <http://www.lal.in2p3.fr/SI/CMT/CMTDoc.html>
 - <http://lhcb.cern.ch/computing/Support/html/cmt.htm>

Anatomy of a requirements file

```
package geomrep
version v1r1

use geometry
use graphics

include_path ${GEOMREPROOT}

macro geomrep_linkopts "-L ${GEOMREPROOT}/${CMTCONFIG} -lgeomrep\"
    visualC ""

macro_append win32Debug " ${GEOMREPROOT}/${DebugDir}/geomreplib.lib"
macro_append win32Release " ${GEOMREPROOT}/${ReleaseDir}/geomreplib.lib"

path_append LD_LIBRARY_PATH "${GEOMREPROOT}/${CMTCONFIG}"

private

library geomrep -no_share *.cxx ../geomrep/*.h

# set up application link-options for test program. Note also the resource
macro_append graphics_linkopts ${graphics_gui_linkopts}
application testgeom -windows test/testgeom.cxx ${win32Resource}
```

Package name;
"used" packages

Compile-time interface
Link interface for unix,
NT clients

Two "constituents":
a library and an
executable

CMT query example and the path

- **cmt show uses**

```
# use geometry *
# use glast_settings *
# use EXTLIB *
# use CLHEP *
# use EXTLIB *
# use graphics *
# use glast_settings *
# use CLHEP *
#
# Selection :
use CMT v1r5p1 d:/packages
use EXTLIB v2 d:/packages/external
use CLHEP v1r5 d:/packages/external
use glast_settings v2
  d:/packages/gismo_v2
use graphics v1r1 d:/packages/gismo_v2
use geometry v1r1 d:/packages/gismo_v2
```

- Were do the explicit paths come from? Ask CMT

- **cmt show path**

```
d:\packages\gismo_v2;d:\packages\external;d:\packages
```

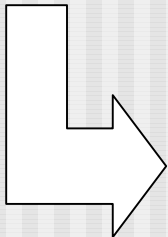
- Flexibility good for development...

- ...but a problem for releases

Building the constituents

- cmt build msdev
 - create Visual Studio files in Visual folder

- make



```
han:/packages/geomrep/v0/mgr[36]$make
/packages/CMT/v1r3/src/Makefile.header:61: cmt_path.make: No such file or directory
/packages/CMT/v1r3/src/Makefile.header:66: setup.make: No such file or directory
/packages/CMT/v1r3/src/constituents.make:5: constituents.make: No such file or directory
/packages/CMT/v1r3/mgr/cmt build make_setup -tag=i586
setup.make:42: i586.make: No such file or directory
constituents.make:9: geomrep.make: No such file or directory
constituents.make:15: testgeom.make: No such file or directory
Now rebuilding testgeom.make
Application testgeom
Now rebuilding geomrep.make
Library geomrep
Now rebuilding i586.make
geomrep.make:35: geomrep_dependencies.make: No such file or directory
testgeom.make:25: testgeom_dependencies.make: No such file or directory
Now rebuilding testgeom_dependencies.make
Now rebuilding geomrep_dependencies.make
echo "";
```



```
i586.make ok
geomrep : library ok
geomrep ok
cd ../i586/; g++ -o testgeom.exe.new ../i586/testgeom.o -L /packages/geomrep/v0/i586 -lgeomrep -L /packages/geometry/
v0/i586 -lgeometry -L /packages/graphics/v0/i586 -lgraphics -L /packages//CLHEP/v1r4/CLHEP -lCLHEP ; mv -f testgeom.
exe.new testgeom.exe
```

What about CVS?

- Version name (e.g. v1r1) corresponds to a *tagged revision*
 - Except if it is being developed
- To checkout a tagged version:
cd geometry
cvs -r co -d v1r1 -r v1r1 cmt/geometry
- Ugly, error-prone.
 - Solution: special script (Masanobu) or user interface (VCMT)

doxygen

- Input: src, <package_name>
- Output: doc/doxy-html
- mainpage convention:
 - <package_name>/mainpage.h

Enter VCMT

- An (NT only) GUI interface to CMT, CVS, doxygen, VisualStudio

Summary of developer actions

| | |
|--------------------------------------|--|
| checkout a package | |
| checkout a group of related packages | |
| create a new package | |
| build a package's constituents | |
| | |
| | |
| | |
| | |